



Fantasy
becomes Reality



プログラマーだってHoudiniを覚えたい！

プログラマーのプログラマーによる プログラマーのためのHoudiniトーク

長船 龍太郎
株式会社ミクシィ
テクニカルアーティスト

鈴木 英樹
YUKE'S LA Inc.
シニアシステムエンジニア

岸川 貴紀
株式会社Cygames
テクニカルアーティスト

山部 道義
株式会社ポリフォニー・デジタル
プログラマー

#cedec_houdini

はじめに



- 概要

Houdini経験豊富なプログラマー・TA同士で
ワイワイ楽しく”**プログラマー視点**”
からのHoudiniトークを行う！

- 対象者

- プログラマー、R&Dエンジニア
- TA (**T**echnical **A**rtist)
- Houdini興味ある方



自己紹介

パネリスト紹介



鈴木 英樹
YUKE'S LA Inc.
シニアシステムエンジニア

- ゲーム→パチンコ→映像→ゲーム
 - ゲーム業界でキャリアをスタート
 - パチンコメーカーを経て、渡米し映像業界へ
 - ゲーム業界にもどり現在はYUKE'S LA Inc.にてR&Dチームに所属、主にグラフィックス周りを担当
 - Houdiniはバージョン12から

パネリスト紹介



岸川 貴紀

株式会社Cygames

テクニカルアーティスト

- 映像 → ゲーム
 - エフェクトアーティスト(映像)を経験
 - テクニカルディレクター(映像)としてエフェクトとパイプラインの開発に携わる
 - Houdiniに関するツール開発の講演を実施
 - 株式会社Cygamesに合流(2018年)
 - ハイエンドゲーム開発プロジェクトに所属
 - エフェクトをメインにサポートしつつ、ツール開発や各種パイプライン開発に従事

パネリスト紹介



山部 道義

株式会社ポリフォニー・デジタル
プログラマー

- 映像→ゲーム→映像→ゲーム
 - 過去にTAとしてSIEに7年ほど在籍
 - Puppeteer/Knack/Knack2/人喰いの大鷲トリコ
 - Houdiniの群衆シミュレーションセミナー登壇
 - 現在はDCCツール関連の開発とパイプライン開発に従事
 - 今回はTA目線で語りたいと思います



長船龍太郎

株式会社ミクシ

XFLAG スタジオ テクニカルアートグループ

- ゲーム → ITコンサル→ゲーム/映像
 - 学生時代は画像認識、プロシージャルの研究に従事
 - プログラマ寄りTAとして様々なプロジェクトで主にアーティスト向け開発環境の整備を行い、外資系ITコンサルティングを経て、ミクシ入社
 - 「ゲーム映像制作パイプライン構築マニュアル」日本語版協力、「海外TA勉強会」「Tokyo Houdini Meetup」発起人
 - 現在、社内Houdini推進プロジェクト遂行中

お題目

- 経緯と目的
- ①プログラマー視点でHoudiniの魅力を語ろう！
- ②業界初！プログラマーによる
Houdiniプロシージャル品評会
- ③仮説検証トーク
～プログラマーはHoudiniと相性がいいのか？
- まとめ～今後のゲーム業界でのHoudiniについて

経緯と目的

経緯～XFLAG スタジオのTA業務

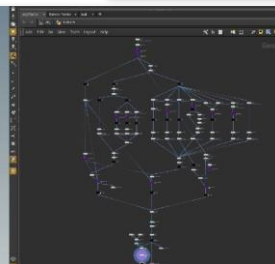
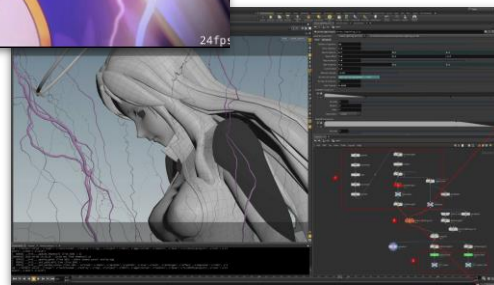
- 様々な事業を横断した技術サポート
 - 作業効率向上には自動化が必須
 - 自動化といえばプロシージャル
 - Houdini導入で全体効率化を模索中
- Houdini導入済
 - アニメ制作、イベント映像
- Houdini未導入
 - 映像制作、スポーツ、Merchandising
 - **ゲーム開発**



経緯～Houdini導入事例

- アニメ モンスターストライク
 - エフェクトはほぼHoudiniで制作
- XFLAG FREAK2018のVJ映像
 - Houdini + TouchDesignerで制作

ゲーム開発で導入できてない！



OMNIBUS JAPAN

経緯～Tokyo Houdini Meetup 開催

- ゲーム開発者向けHoudiniイベント
- 100名を超える参加者で大盛況！
- 参加者の**4割弱**がプログラマーor TA/TD
 - 興味持ってる方
 - 独学してる方
 - 既に業務で導入してる方…

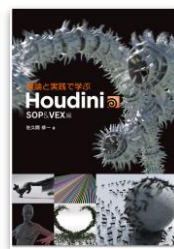


プログラマはHoudiniと相性が良いのではないか
という仮説が浮かび始める…

経緯～社内Houdini勉強会開始！

- 2018年2月から毎週開催中
- 様々なメンバーが参加中
 - 映像制作アーティスト
 - ゲームエフェクトアーティスト
 - テクニカルアーティスト
 - サウンドクリエイター
 - アニメ制作アーティスト …

プログラマーがいない…！



目的

- Houdiniコミュニティにプログラマも巻き込みたい
 - プログラマーがHoudiniに興味を持つきっかけにする
 - Houdini導入促進のきっかけにする
- 仮説の検証を行いたい

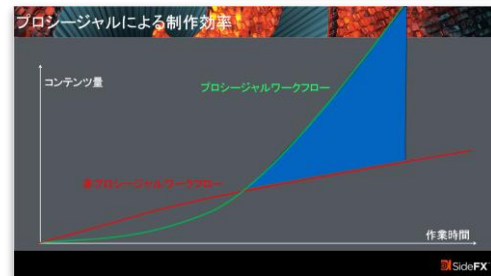
**プログラマーはHoudiniと
相性が良いのではないか**

①プログラマー視点で Houdiniの魅力を語ろう！

Houdini導入のきっかけは？ (1/3)

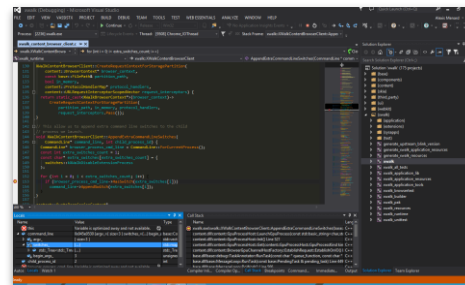
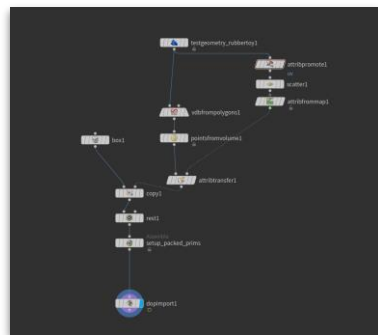
- なぜ Houdiniに興味を持ったのか？
- どのようにして Houdiniを学習したか？
- どのようにして 実業務に導入したか？
- キーワード
 - プロシージャル、自動化、作業効率向上
 - Houdini Apprentice (無料版)、Houdini学習曲線
 - SideFX多喜さん、身近なHoudini詳しい人

Houdini



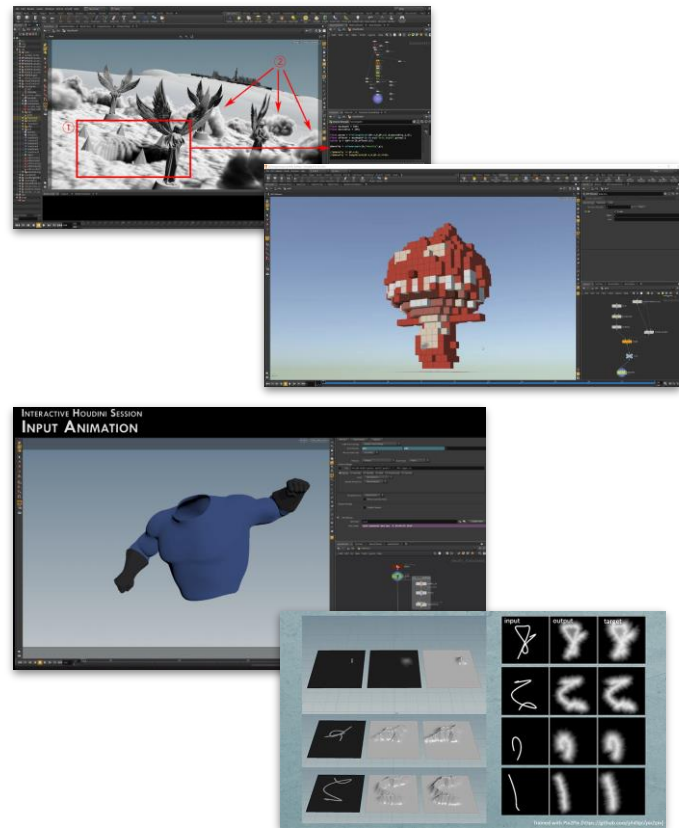
ノードベースのワークフロー (2/3)

- レゴブロック感覚で積み上げていく
 - 1ノード≒1スニペット (最小単位の機能、マイクロベース)
- 標準ノードの組み合わせで大抵のことはできる
 - 足りない部分はカスタマイズ (VEX, Python, SDK, Expression)
- データプロセスを確実に追える (シーン解析、ドキュメント参照)
 - デバッガでソースを追うアプローチと似ている



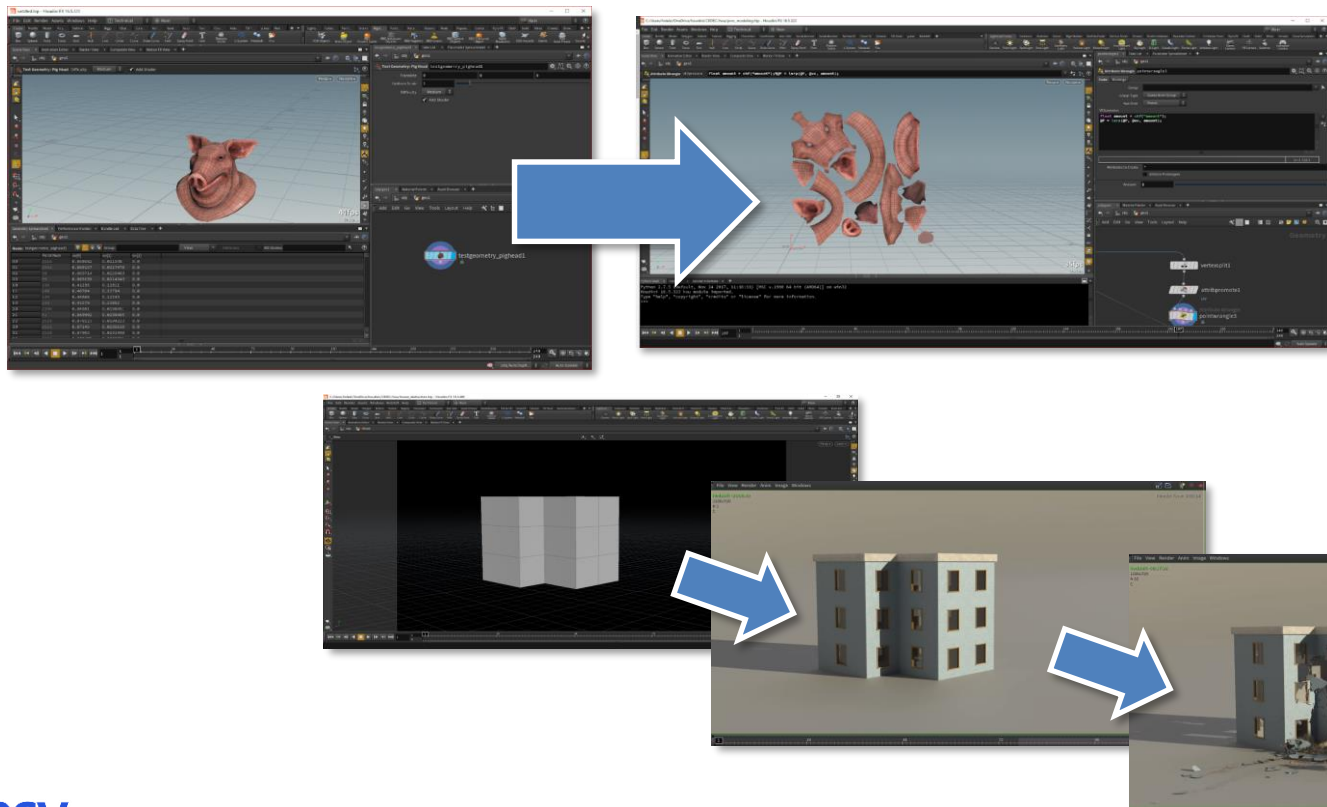
R&Dのプラットフォーム (3/3)

- Houdiniを利用した研究事例
 - PixarのDynamic Kelvinletsの解説
 - AI（機械学習・深層学習）と絡めた研究も
- 新技術の検証窓口として
 - ランタイム実装前に高速な試行錯誤が可
- CG技術の先生として
 - ゲーム外のCG技術をいち早く触られる
 - プログラマ視野が広がりキャリアアップ！

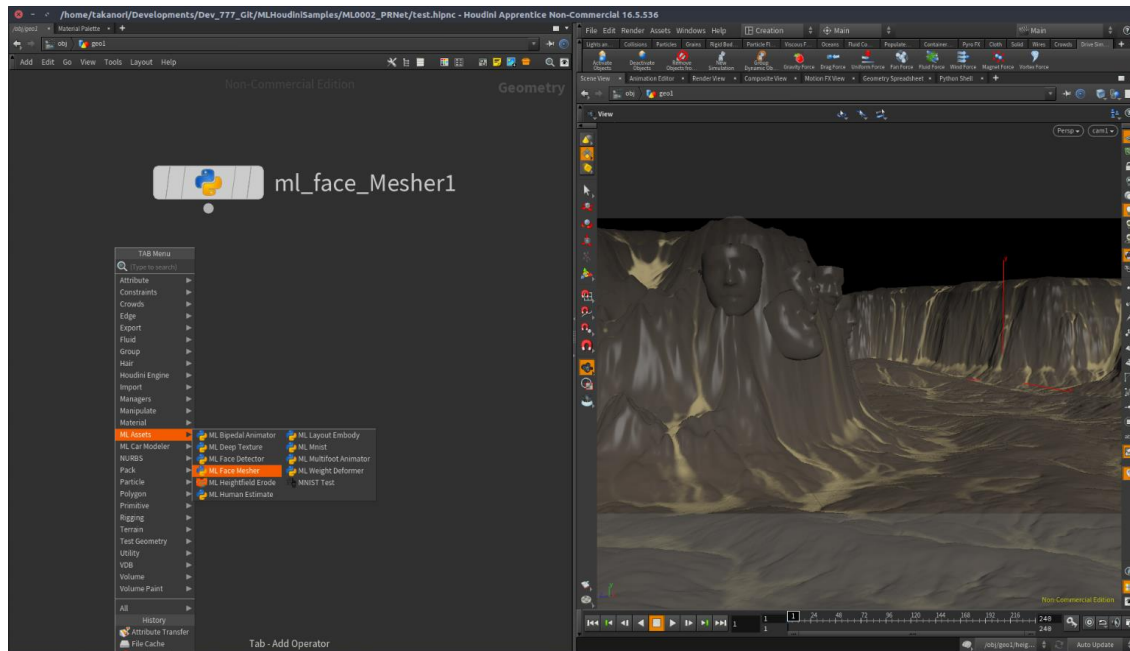
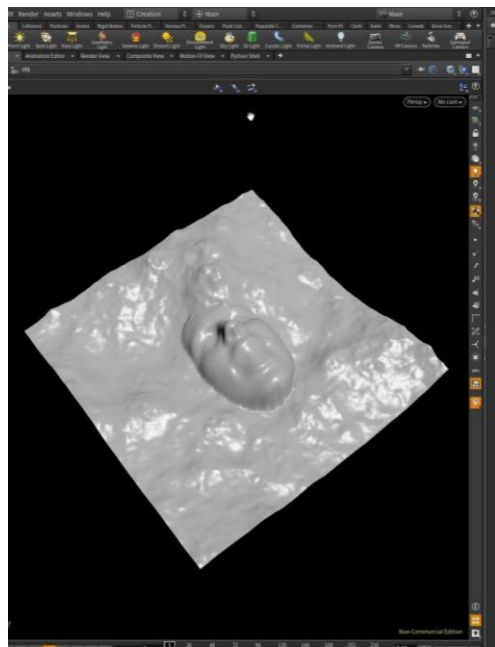


②業界初！プログラマーによる Houdiniプロシージャル品評会

鈴木さんPCに切り替えます

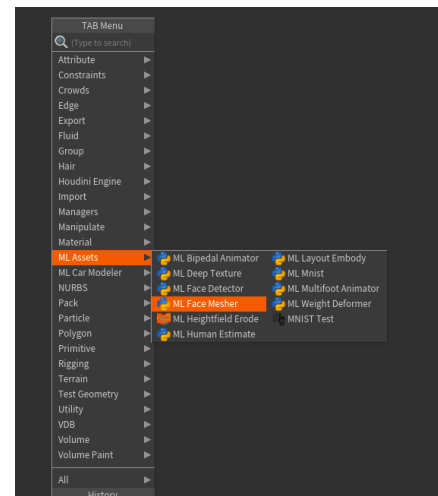


機械学習を使ったノードの開発



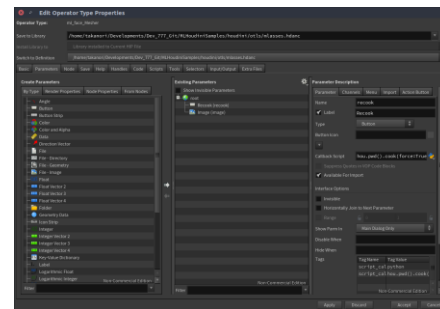
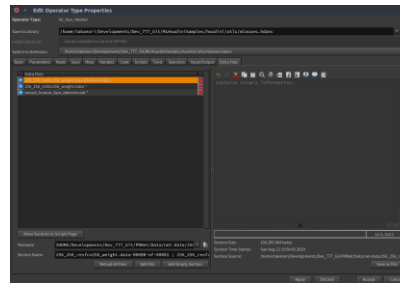
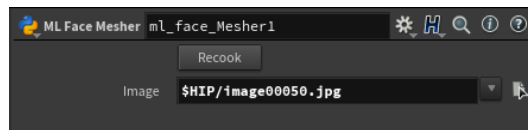
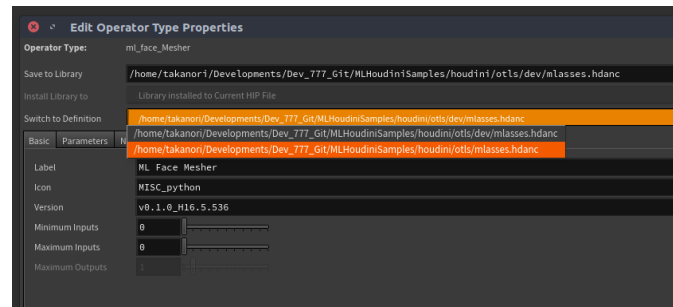
Houdiniを機械学習のプラットフォームにするメリット

- レゴブロック的な研究・開発フロー
 - レゴブロックのブロックを作ること、組み上げていくという感じ
 - プログラマやリサーチャーはノード(=ブロック)を作る
 - アーティストはそのノードを利用して絵作りをする
- Houdini Digital Asset(HDA)はコードのスニペット
 - 資産として蓄積可能
 - かつ、現場で即座に使ってもらえる環境



HDAはスニペット

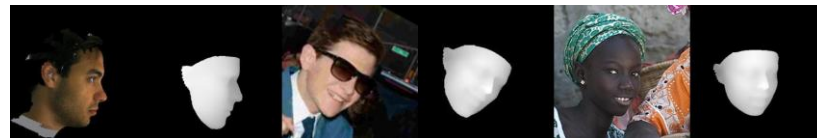
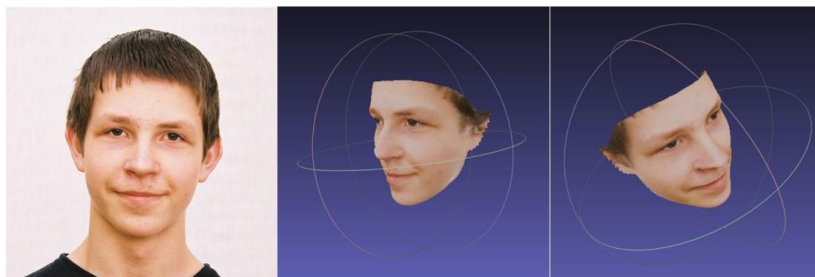
- HDAは超高機能スニペット
 - D&Dで入力UIを作れる
 - 同名のノードの切り替え機能
 - リリースVer. ⇔ 開発 Ver.
 - バイナリなどのExtra Fileを持てる
 - 学習済みパラメーターファイルとか
- Asciiでもバージョン管理出来る
 - Gitでも管理可能



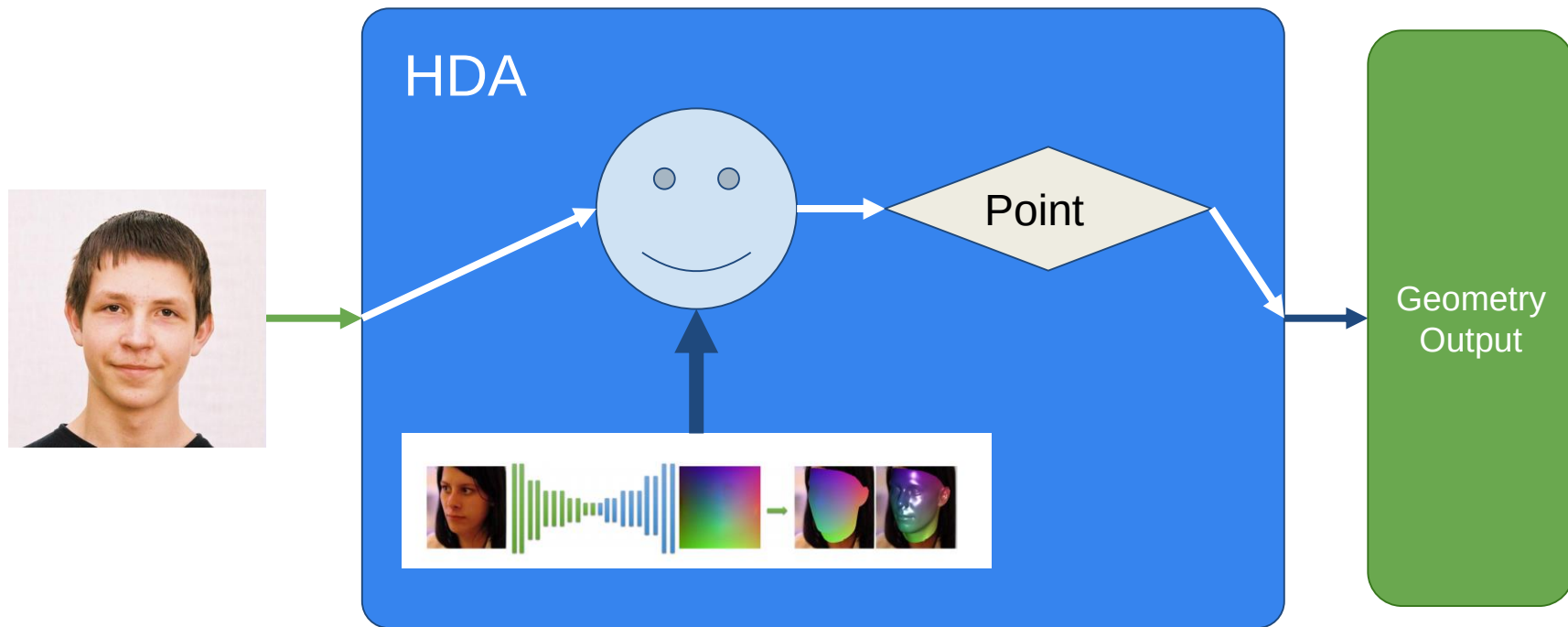
PRNetのMeshing部分を実装してみる

PRNetとは

- 2D->3D化の推論を行う機械学習ライブラリ
 - 顔の形状を元に頂点情報(及び色情報)を推測
 - 推測された頂点情報を元にメッシュ生成
- GitHubにて公開されている



HDAへの推論処理の実装



APIを使うとこの要点

```
import tensorflow as tf
```

```
from PRN import api
```

~~~

```
geo = hou.pwd().geometry()
```

~~~

```
with tf.Graph().as_default():
```

```
    prn = api.PRN(is_dlib=True)
```

```
    pos = prn.process(image)
```

```
    vertices = prn.get_vertices(pos)
```

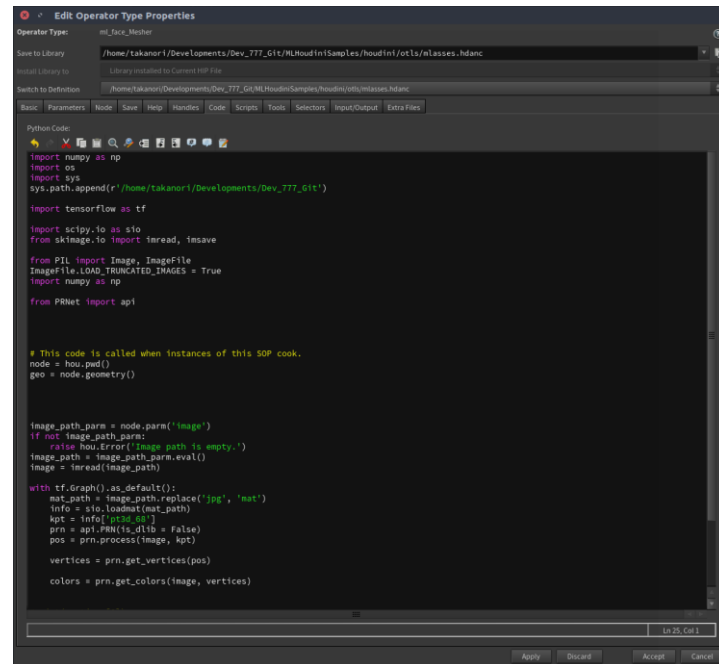
~~~

```
for vert in vertices:
```

```
    vertvec = vert.tolist()
```

```
    point = geo.createPoint()
```

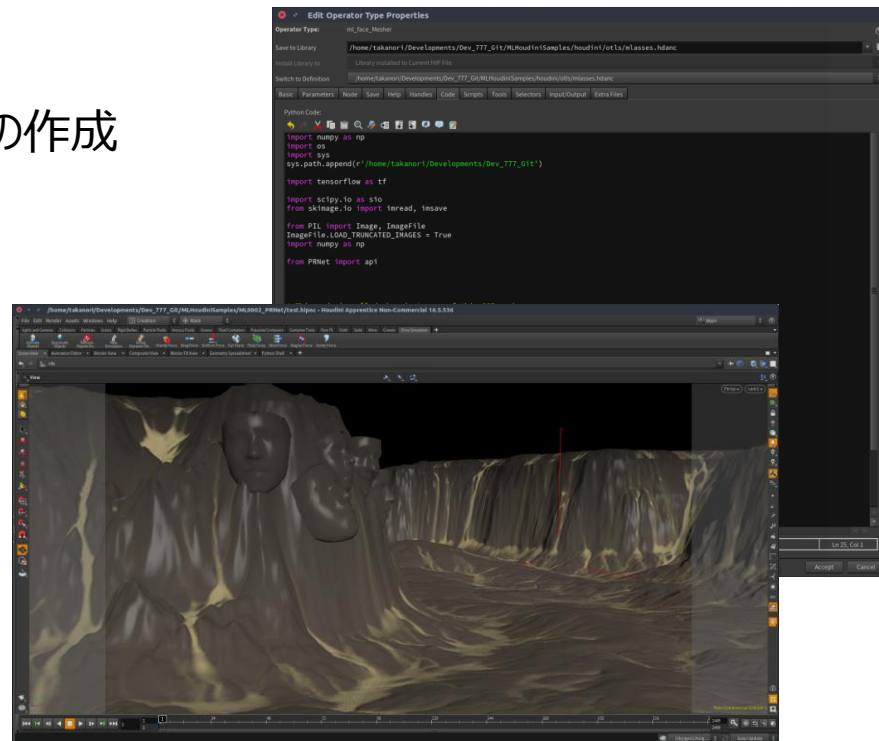
```
    point.setPosition(vertvec)
```



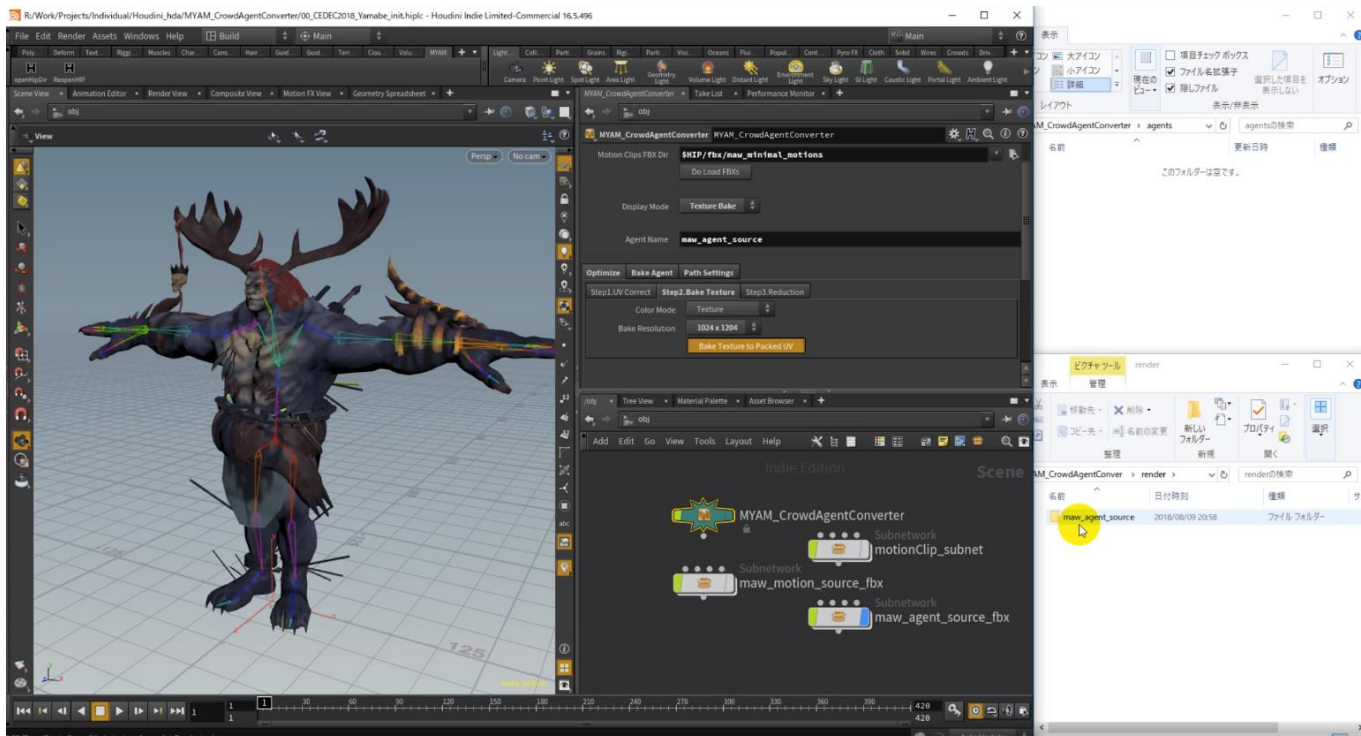
# 機械学習ノードを作る手順

1. モジュールの環境を作る
2. データセットからの学習を行う
3. File > New Assets...で新しいAssetの作成
4. Codeタブを開く
5. コードを書く
  - a. モジュールのインポート
  - b. 学習モデルからの推論処理
  - c. 結果をPoint Attributeに反映

割とこれだけ！



# 群衆向けキャラクターリダクションHDA

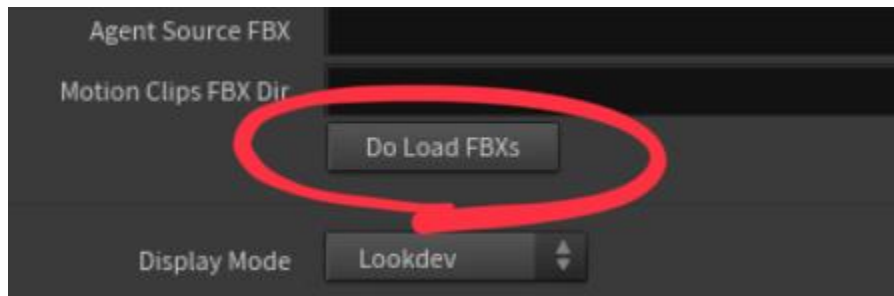


- **必要な処理**
  - a. 各種FBXデータ読み込み/ジオメトリ統合
  - b. Lookdev
  - c. UVパック
  - d. テクスチャバイク
  - e. ジオメトリのリダクション
  - f. キャプチャ情報の転送
  - g. アニメーションの転送
  - h. エージェントの自動バイク
  - i. データの微調整/再バイク

※時間的な都合で全工程を詳しく説明できないので、かいつまんでご紹介します ([動画](#))

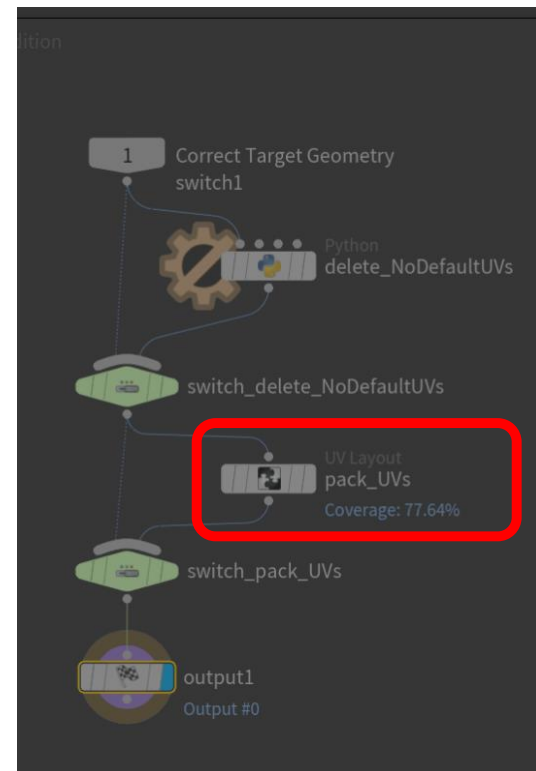
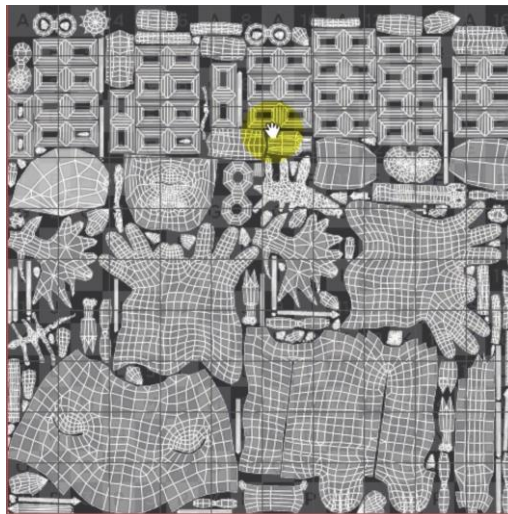
# 各種FBXデータ読み込み/ジオメトリ統合

- HDA内のボタンからPythonスクリプトを実行
  - FBXのロード
  - FBX内の各ジオメトリを一つのGeometryに収集
  - 新規ベイクマテリアルの作成とアサイン
    - FBXマテリアルからLookdev用のPrincipled Shaderを作成
    - テクスチャベイク用のPrincipled Shaderを作成



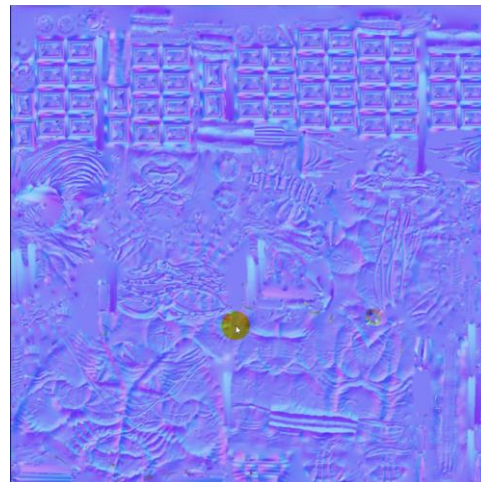
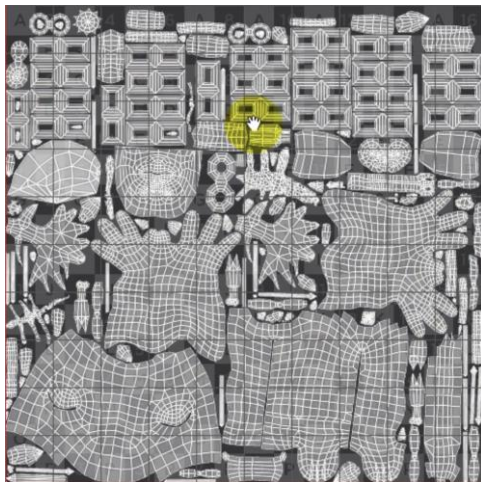
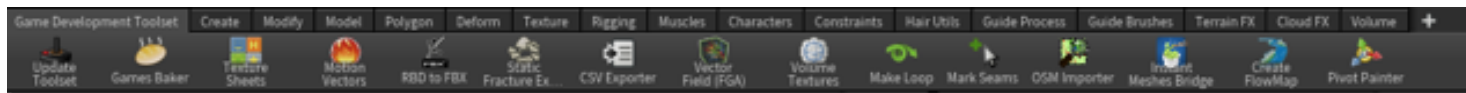
# UVパック

- UV Layout SOP
  - UVパックを行う



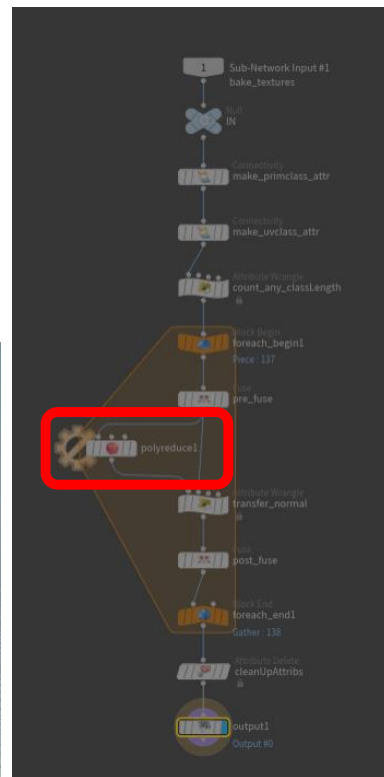
# テクスチャバイク

- Games Baker ROP(Game Development Toolset)
  - 異なるジオメトリ間でテクスチャを転写



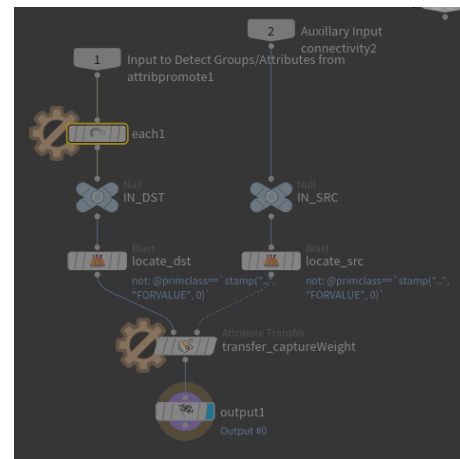
# ジオメトリのリダクション

- Poly Reduce SOP
  - ポリゴンリダクションを行う
    - 強めのリダクションでもシルエットを保持



# キャプチャ情報の転送

- Attribute Transfer SOP
  - 異なるジオメトリのコンポーネント間で、距離をもとにアトリビュートを転送
- リダクション前モデルからリダクション済みモデルへ@boneCaptureを転送
  - 事前にシェル単位でジオメトリにIDを付与
  - リダクション前後のジオメトリをIDごとに抽出
  - リダクション前後のペアジオメトリ間のみでウェイトを転送することで精度を担保



- **必要な処理**

- a. **各種FBXデータ読み込み/ジオメトリ統合** (カスタムスクリプト)
- b. Lookdev (ビルトインノード)
- c. **UVパック** (ビルトインノード)
- d. **テクスチャベイク** (GameDevelopmentToolSet)
- e. **ジオメトリのリダクション** (ビルトインノード)
- f. **キャプチャ情報の転送** (カスタムノード+ビルトインノード)
- g. アニメーションの転送 (ビルトインノード)
- h. エージェントの自動ベイク (ビルトインノード)
- i. データの微調整/再ベイク (ビルトインノード)

**最小限の開発コスト**

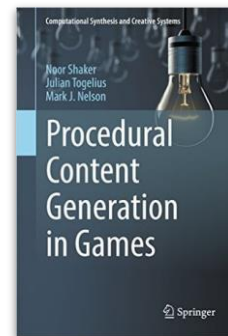
# ③ 仮説検証トーク ～プログラマーはHoudiniと 相性が良いのではないか

# プログラマーがHoudiniと相性が良い点

- プログラム知識はHoudini習得をブーストする
  - 論理的思考、ノードベース、プログラム言語に協調性がある
  - HDAを使い回したり拡張する思考はプログラミングそのもの
- AI（機械学習、深層学習）の実装効率性が高い
- プログラマとアーティストの垣根が低くなる
  - 「アーティストの感覚」を言語化することで論理的に解析
  - アーティストとのコミュニケーション促進につながる

# 注意すべき点（懸念点）

- 沼にハマると…他のDCCツールよりも  
工数や管理コストがかかる可能性がある
  - 明確なHoudini活用の目的がないと危険
  - 管理して導くHoudini TD/TAの存在が重要
- やはり学習コストはかかる
  - 同士（コミュニティ）を作って共に学ぶのが重要
  - チュートリアルは最新のものからやろう



**プログラマーはHoudiniと  
相性が良いに違いない**

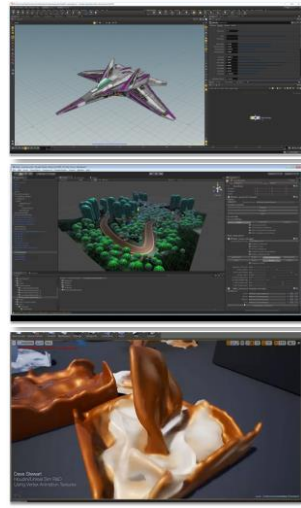
# まとめ

## ～今後のゲーム業界における Houdiniについて考える

# 今後のゲーム業界におけるHoudini



- Houdiniの柔軟さをふんだんに活かそう
  - プロシージャルモデリング（ジオメトリ最適化、LOD、バリエーション）
  - レベルデザインの効率化（地形生成、アセット自動配置）
  - 高度なエフェクト・シミュレーション表現（Vertex Animation Texture）
- 情報量増加 vs 人的リソースに備えよう
  - 堅牢なデータ作成のためニーズが高まると予想
- 問題解決したい事・試したい技術をどんどん試そう
- 全体絵の見た目（アート）のセンスも磨こう



**プログラマが加わることで  
Houdiniコミュニティはより活性化する！**

# ご清聴ありがとうございました！



ご清聴  
ありがとうございました



# Houdiniコミュニティの紹介

- Facebookグループ「Tokyo Houdini Meetup」



# ご質問はありますか？

